PORTAL THE ACM DIGITAL LIBRARY

> home : > about : > feedback : > login

US Patent & Trademark Office

Try the *new* Portal design
Give us your opinion after using it.

Search Results

Search Results for: **[private<AND>((global <near> breakpoint<AND>((global and breakpoint))))]**
Found **58** of **127,944 searched.**

Search within Results

> Advanced Search

> Search Help/Tips

Sort by: Title Publication Publication Date **Score** Binder

Results 1 - 20 of 58 short listing

Prev Page 1 2 3 Next Page

1 A retargetable debugger 82%
Norman Ramsey , David R. Hanson
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1992 conference on Programming language design and implementation July 1992
Volume 27 Issue 7
We are developing techniques for building retargetable debuggers. Our prototype, 1db, debugs C programs compiled for the MIPS R3000, Motorola 68020, SPARC, and VAX architectures. It can use a network to connect to faulty processes and can do cross-architecture debugging. 1db's total code size is about 16,000 lines, but it needs only 250-550 lines of machine-dependent code for each target. 1db owes its retargetability to three techniques: getting help from the compiler, usin ...

2 Fast detection of communication patterns in distributed executions 80%
Thomas Kunz , Michiel F. H. Seuren
Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research November 1997
Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

3 Debuggable concurrency extensions for standard ML 80%
Andrew P. Tolmach , Andrew W. Appel
ACM SIGPLAN Notices , Proceedings of the 1991 ACM/ONR workshop on Parallel and distributed debugging December 1991

Volume 26 Issue 12

4 A program debugger for a systolic array: design and implementation 77%
Bernd Bruegge , Thomas Gross
ACM SIGPLAN Notices , Proceedings of the 1988 ACM SIGPLAN and SIGOPS workshop on Parallel and distributed debugging November 1988
Volume 24 Issue 1

The Warp machine consists of a programmable linear systolic array connected to a general-purpose workstation host. Warp can be accessed either locally from this host or remotely from a large number of workstations connected to a local area network. Since the linear arrangement of the cells in the array restricts direct input and output with the host to the boundary cells, a source language debugger is important for program development. The Warp debugger is integrated into the Warp Programmi ...

5 Supporting reverse execution for parallel programs 77%
Douglas Z. Pan , Mark A. Linton
ACM SIGPLAN Notices , Proceedings of the 1988 ACM SIGPLAN and SIGOPS workshop on Parallel and distributed debugging November 1988
Volume 24 Issue 1

Parallel programs are difficult to debug because they run for a, long time and two executions may yield different results. Reverse execution, is a simple and powerful concept that solves both these problems. We are designing a tool for debugging parallel programs, called Recap, that provides the illusion of reverse execution using checkpoints and event recording and playback. During normal execution, Recap logs the results of system calls and shared memory reads: as well as ...

6 Fast breakpoints: design and implementation 77%
Peter B. Kessler
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1990 conference on Programming language design and implementation June 1990
Volume 25 Issue 6

We have designed and implemented a fast breakpoint facility. Breakpoints are usually thought of as a feature of an interactive debugger, in which case the breakpoints need not be particularly fast. In our environment breakpoints are often used for non-interactive information gathering; for example, procedure call count and statement execution count profiling [Swinehart, et al.]. When used non-interactively, breakpoints should be as fast as possible, so as to perturb the execution of the pro ...

7 Program development for a systolic array 77%
Bernd Bruegge
ACM SIGPLAN Notices , Proceedings of the ACM/SIGPLAN conference on Parallel programming: experience with applications, languages and systems January 1988
Volume 23 Issue 9

The primary objective of the Warp programming environment (WPE) is to simplify the use of Warp, a high-performance programmable linear systolic array connected to a general-purpose workstation host. WPE permits the development of distributed applications that access Warp either locally from the host or remotely from a large number of workstations connected to a local area network. Its audience includes the user who calls routines from a library, the programmer who develops new algorithms fo ...

8 Experiences with building distributed debuggers 77%
Michael S. Meier , Kevan L. Miller , Donald P. Pazel , Josyula R. Rao , James R. Russell

Proceedings of the SIGMETRICS symposium on Parallel and distributed tools January 1996

9 A framework for providing consistent and recoverable agent-based access to heterogeneous mobile databases 77%
Evaggelia Pitoura , Bharat Bhargava
ACM SIGMOD Record September 1995
Volume 24 Issue 3

Information applications are increasingly required to be distributed among numerous remote sites through both wireless and wired links. Traditional models of distributed computing are inadequate to overcome the communication barrier this generates and to support the development of complex applications. In this paper, we advocate an approach based on agents. Agents are software modules that encapsulate data and code, cooperate to solve complicated tasks, and run at remote sites with minimum inter ...

10 K9: a simulator of distributed-memory parallel processors 72%
P. Beadle , C. Pommerell , M. Annaratone
Proceedings of the 1989 ACM/IEEE conference on Supercomputing August 1989

K9 is a software package for the simulation and performance evaluation of distributed-memory parallel processors (DMPPs). It is written in C++ and runs on Sequent Symmetry and SUN-3. K9 provides the user with four building-blocks (processor cells, communication channels, multi-port shared-memories, and I/O processors), and one abstraction mechanism (the DMPP interconnection topology). Application code for K9 can be written in C++ or C. When timing analysi ...

11 A new framework for debugging globally optimized code 70%
Le-Chun Wu , Rajiv Mirani , Harish Patil , Bruce Olsen , Wen-mei W. Hwu
ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1999 conference on Programming language design and implementation May 1999
Volume 34 Issue 5

With an increasing number of executable binaries generated by optimizing compilers today, providing a clear and correct source-level debugger for programmers to debug optimized code has become a necessity. In this paper, a new framework for debugging globally optimized code is proposed. This framework consists of a new code location mapping scheme, a data location tracking scheme, and an emulation-based forward recovery model. By taking over the control early and emulating instructions selective ...

12 Progress report: Brown university instructional computing laboratory 67%
Marc H. Brown , Robert Sedgewick
ACM SIGCSE Bulletin , Proceedings of the fifteenth SIGCSE technical symposium on Computer science education January 1984
Volume 16 Issue 1

An instructional computing laboratory, consisting of about 60 high-performance, graphics-based personal workstations connected by a high-bandwidth, resource-sharing local area network, has recently become operational at Brown University. This hardware, coupled with an innovative courseware/software environment, is being used in the classroom in an attempt to radically improve the state of the art of computer science pedagogy. This paper describes the current state of the project. T ...

13 Automating the re-declaration of unneeded globals as private 56%

Amitava Datta , Prabhaker Mateti
Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice March 1993

14 Experiences developing and using an object-oriented library for program manipulation 54%
Tim Bingham , Nancy Hobbs , Dave Husson
ACM SIGPLAN Notices , Proceedings of the eighth annual conference on Object-oriented programming systems, languages, and applications October 1993
Volume 28 Issue 10

15 Report on the seventh ACM SIGOPS European workshop: systems support for worldwide applications 49%
Andrew S. Tanenbaum
ACM SIGOPS Operating Systems Review January 1997
Volume 31 Issue 1

16 Integrating and customizing heterogeneous e-commerce applications 48%
Anat Eyal , Tova Milo
The VLDB Journal — The International Journal on Very Large Data Bases August 2001
Volume 10 Issue 1

A broad spectrum of electronic commerce applications is currently available on the Web, providing services in almost any area one can think of. As the number and variety of such applications grow, more business opportunities emerge for providing new services based on the integration and customization of existing applications. (Web shopping malls and support for comparative shopping are just a couple of examples.) Unfortunately, the diversity of applications in each specific domain and the dispar ...

17 Features: Code Spelunking: Exploring Cavernous Code Bases 45%
George V. Neville-Neil
Queue September 2003
Volume 1 Issue 6

Try to remember your first day at your first software job. Do you recall what you were asked to do, after the human resources people were done with you? Were you asked to write a piece of fresh code? Probably not. It is far more likely that you were asked to fix a bug, or several, and to try to understand a large, poorly documented collection of source code.

Of course, this doesn't just happen to new graduates; it happens to all of us whenever we start a new job or look at a ...

18 A parallel computer for lattice Gauge theories 43%
T. W. Chiu
Proceedings of the third conference on Hypercube concurrent computers and applications: Architecture, software, computer systems, and general issues - Volume 1 January 1988

A parallel computer especially designed for the calculations in lattice gauge theories is proposed. The final device will contain 256 nodes running in MIMD mode with a computational power about one billion 32-bit floating point operations per second. Each node is controlled by the Motorola 68020/68882 microprocessors, (the Weitek floating-point processors) and two megabyte static RAM. The architecture is designed to allow rapid execution of the numerically intensive calculations in lattice ...

19 The structure of Cedar 33%
Daniel C. Swinehart , Polle T. Zellweger , Robert B. Hagmann
Proceedings of the ACM SIGPLAN 85 symposium on Language issues in programming environments June 1985
Volume 20 , 18 Issue 7 , 6

This paper presents an overview of the Cedar programming environment, focusing primarily on its overall structure: the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. We will emphasize the extent to which the Cedar language, with runtime support, has influenced the organization, comprehensibility, and stability of Cedar. Produced in the Computer Science Laboratory (CS ...

20 A structural view of the Cedar programming environment 27%
Daniel C. Swinehart , Polle T. Zellweger , Richard J. Beach , Robert B. Hagmann
ACM Transactions on Programming Languages and Systems (TOPLAS) August 1986
Volume 8 Issue 4

This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

Results 1 - 20 of 58 short listing

Prev Page 1 2 3 Next Page